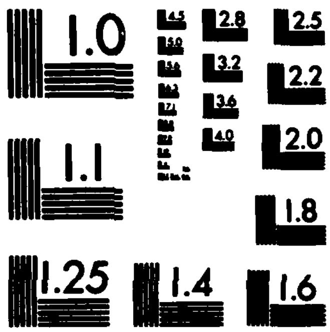
AD-A132 523 ARGOT: A SYSTEM OVERVIEW(U) ROCHESTER UNIV NY DEPT OF COMPUTER SCIENCE J F ALLEN APR 82 TR-101 N00014-82-K-0193 1/1

UNCLASSIFIED F/G 9/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

A132523

ARGOT: A System Overview

James F. Allen
Computer Science Department
University of Rochester
Rochester, NY 14627

TR 101
April 1982

Department of Computer Science
University of Rochester
Rochester, New York 14627

ARGOT: A System Overview

James F. Allen
Computer Science Department
University of Rochester
Rochester, NY 14627

TR 101
April 1982

Abstract

We are engaged in a long-term research project that has the ultimate aim of describing a mechanism that can partake in an extended English dialogue on some reasonably well specified range of topics. The fundamental assumption in this project is that conversants in a dialogue are constantly recognizing and monitoring the goals of the other participants. To do this, they must have a rich body of knowledge about the topic, about the goals and beliefs of the other participants, and about the structure of dialogues in general.

This paper describes progress made towards these goals and outlines the current research areas in which the project is focused. It describes the basic theory underlying our work and the initial system built according to this theory. It then considers some deficiencies in this system and describes the new system currently under development. Finally, various specific research efforts within the group are described.

The preparation of this paper was supported in part by the National Science Foundation under Grant IST-8012418, the Office of Naval Research under Grants N00014-80-C-0197 and N00014-82-K-0193, and the Defense Advanced Research Projects Agency under Grant N00014-78-C-0164.

1. Background

Most current natural language understanding systems do not engage in a dialogue in any general sense. The "conversations" with these systems consist of a series of single question/answer pairs that are analyzed without any consideration of the user's overall goals. Knowledge of the inter-relations between succeeding questions is very limited, typically providing a mechanism for resolving anaphoric reference and possibly some forms of ellipsis. There is no sense of a continuing interaction in which a topic is developed and tasks are accomplished.

Some story comprehension systems (e.g., [Bruce and Newman, 1978; Wilensky, 1978; Carbonell, 1978]) analyze the intentions of characters in the story being understood, and answer questions about these characters' goals. But these techniques are not used to analyze the questioner's intent, or to make the system an active participant in the question answering dialogue that tests the system's comprehension of the story.

Consider Dialogue 1, a sample fragment of a dialogue that serves to motivate our work. This is a slightly cleaned up version of an actual dialogue between a computer operator and a user communicating via terminals.

---

(1) User: Could you mount a magtape for me?
(2) It's tape xxx.
(3) No ring, please.
(4) Can you do it in five minutes?

(5) System: Sorry, we are not allowed to mount that magtape, you will have to talk to [Operator yyy] about it.

(6) User: How about tape zzz?

Dialogue 1.

There are many things the system (acting as the operator) must be able to infer. For instance, the first utterance, taken literally, is a query about the system's abilities. In this dialogue, however, the user intends it as part of a request to mount a particular magtape. Utterance (2) identifies the tape in question, and the third and fourth add constraints on how the requested mounting is supposed to be done. These four utterances, taken as a unit, can be summarized as a single request to mount a particular magtape with no ring within five minutes.

Furthermore, once the above is inferred, the system generates an answer that not only denies the request but provides additional information that may be helpful to the user. The operator believes that talking to the other operator will be of use to the user because he has recognized the user's goal of getting a tape mounted. Utterance (6) taken in isolation is meaningless; however, in the context of the entire dialogue, it can be seen as an attempt to modify the original request by changing the tape to be mounted.

Another problem facing the system is deciding when to speak. In another dialogue the user might not have provided the additional information (such as whether to use a ring) in later utterances, and the system would have had to ask the user for clarification.

We are currently building a system that provides some answer to each of the above difficulties. It is based on the following assumptions:

-- The participants in the dialogue are both goal-directed reasoning systems that can perform physical actions including linguistic communication and mental actions such as inference.

-- Language arises in an attempt to achieve some goal (e.g., obtain information, get the other to do some task).

-- Each participant attempts to understand the other's utterances by recognizing the goals that motivated them. They mutually develop a common base of knowledge about the task under discussion as the dialogue progresses.

-- Cooperation between the participants occurs when one participant accepts a goal of the other as his or her own goal.

In order to develop this model further we need to investigate the nature of the goals and actions in such a setting. This is not the place to examine such issues in detail (see [Allen and Perrault, 1980]), but a brief summary is necessary to understand the remainder of the paper.

Most goals in this setting involve acquiring beliefs and influencing other's beliefs and goals. These goals are typically achieved using linguistic actions (speech acts) such as *informing*, *requesting*, *warning*, etc. Speech acts are defined by specifying the prerequisites and effects which typically are conditions on the beliefs of the speaker and hearer.

To give an idea of the necessity for this analysis, consider a set of situations in which two agents, S and H, discuss a secret. The situations differ only in what the agents know about each other's knowledge of the secret. In each, we shall consider the plausible interpretations of the utterance "Do you know the secret?"

Setting 1: If S knows the secret and believes that H doesn't know the secret, then "Do you know the secret?" is probably an *offer* to tell H the secret.

Setting 2: If S doesn't know the secret and believes that H does know the secret, then "Do you know the secret?" is probably a *request* that H tell S the secret.

Setting 3: If S knows the secret and doesn't know if H knows the secret, then "Do you know the secret?" is probably either a literal yes/no question or a conditional *offer* to tell H the secret.

The only changes in the above settings involved S's and H's beliefs about each other. The interpretations of the utterance arise from considering what goals are plausible given what S and H know about each other.

Formalizing adequate models of belief and action is a difficult task, but initial attempts have been made (e.g., [Moore, 1979; Allen and Perrault, 1980]) that provide a basis for future work. Our recent efforts in this area will be discussed later in the paper.

2. A Simple Dialogue Model

Given this background, I can now describe a simple model of a participant in a dialogue. This model was implemented in a system that simulated a clerk in an information booth in a train station [Allen, 1979]. Once this system is described, we can examine its inadequacies and thus motivate the discussion of the current system.

The model uses the above theory and outlines four major steps in modeling a participant. These are:

1) Identify the linguistic actions performed by the speaker using syntactic and semantic analysis, taking the utterance *literally*.
2) Recognize at least part of the speaker's plan by finding an inference path connecting the observed linguistic action(s) to an expected goal in the context.
3) Choose a set of goals by identifying the key steps in the other's plan that cannot be achieved without assistance (i.e., the *obstacles*).
4) Plan a response that achieves the goals identified in Step (3).

In the train station dialogues, the goals of the users were assumed to be one of the following:

-- boarding a train;
-- meeting an arriving train;
-- other (chosen only if above two are eliminated)

Let us consider it operating on the simple question

"When does the Montreal train leave?"

In Step (1), this was analyzed to be an instance of the action

User REQUEST that
    System INFORM user of the departure time.

A simple outline of the plan recognized in Step (2) is as follows:

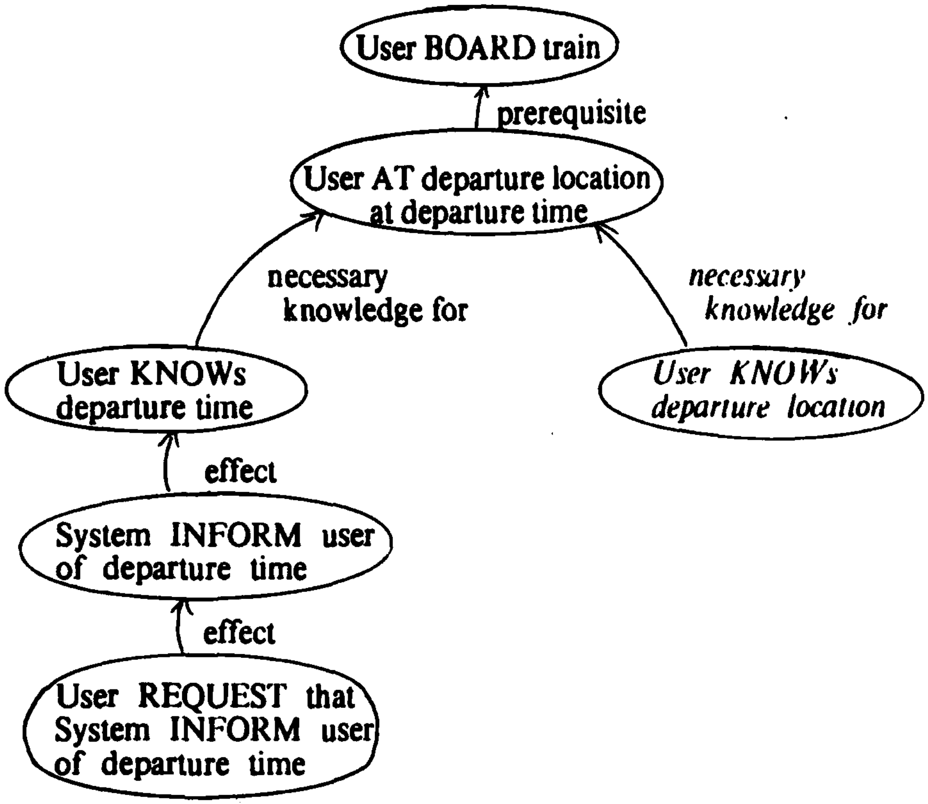


Figure 1: A Simple Plan Recognized from
"When does the Montreal train leave?"

Reading the plan from the bottom to the top, we see the following connections. An eventual effect of the user's REQUEST is that the system performs the requested action, namely the INFORM. The effect of the INFORM action is that the user will KNOW the departure time. This knowledge is necessary for the user to achieve the goal of being at the departure location at the departure time, which in turn is a prerequisite for boarding the train. Since boarding the train is an expected goal in this context, we are done.

In Step (3), the system examines the user's plan and finds two obstacles. The first was directly on the path outlined above: the user needs to KNOW the departure time. The second is implicit from general knowledge about the structure of plans: the user also needs to know the departure location. If the context were slightly different, say the station had only one track, then the system would have believed that the user already knew the departure location, and thus it would not be an obstacle. In this context, however, the system believes that users do not generally know this information. The system's response from Step (4) addresses both these goals, and the answer is:

"4:00 at gate 7."

Thus we have seen how a helpful response can be generated. The exact same mechanism can also account for comprehending many indirect speech acts as well as simple noun phrase sentence fragments.

The following two short dialogues give an indication of these abilities:

User: The 3:15 train to Windsor?

System: Gate 10

Dialogue 2: A Simple Noun Phrase.

Here the only reasonable plan in the context that involved such a train was the boarding plan. The answer was generated from the obstacles detected in the plan.

User: Do you know when the Rapido leaves?

System: 4:20.

Dialogue 3: A Simple Indirect Speech Act.

The most important point to remember here is that the user's plan was recognized starting from the literal interpretation of the utterance. The indirect interpretation falls out of the plan analysis (see [Perrault and Allen, 1980] for more details).

3. The Current System

In the current system we are extending the previous work in a number of ways. Most importantly, the earlier model had no knowledge of discourse structure, so could not partake in an extended dialogue. The only constraints on what was said arose from the structure of the plans that were constructed. Also, the parsing model was too weak to analyze any fragments more complicated than simple noun phrases. Many sentence fragments are considerably more complex than this. Finally, the theoretical work on the formal models of belief, action, goals, and plans needed strengthening.

The architecture of the current system can be motivated best by considering the first problem introduced above. Consider the beginning of Dialogue (1):

User: Could you mount a magtape for me?
It's tape xxx.

The first of these utterances can be analyzed in the old system. Let us assume it is recognized as an indirect request and that the user's goal is to get a magtape mounted. What is the user's goal in the second utterance? From one viewpoint, it is still to get the tape mounted. From another viewpoint, however, the important goal to recognize is that this sentence is intended to elaborate on the previous request, i.e., it is specifying the value of a parameter in the plan that was recognized from the previous utterance. The goals at this level of analysis are only indirectly related to the goal of mounting the tape. Thus we find that there are at least two levels of goal analysis that must be considered. Recognition of intention then proceeds at both these levels of analysis. Note that a similar need to recognize goals at different levels has been identified when understanding stories involving conversations (e.g., [Johnson and Robertson, 1981]).

The two levels that we have identified are the *task level*, which includes goals such as mounting tapes, restoring files, etc., and the *communication level*, which includes such goals as introducing a topic, clarifying or elaborating on a previous utterance, modifying the current topic, etc. In the dialogues we consider, the topics generally concern some task that the user needs assistance in performing.

Given this distinction, we can see where other recent dialogue systems fit into this framework. The work at SRI [Walker, 1978] in the expert-apprentice dialogues monitored the goals of the user at the task level. The only analysis at the communicative goal level was implicit in various mechanisms such as the focusing of attention [Grosz, 1978]. This work ties the task structure and communicative structure too closely together for our purposes.

The work of Mann et al. [1977] and Reichman [1978] both can be seen as analyses of the communicative goals underlying sentences. Thus these give a clue to the set of high-level goals in the communicative goal plan recognition. Neither of these analyses describe in detail the process of recognizing the communicative goals from actual utterances.

The system described in Section 2 and the work at BBN [Brachman, 1979] have both levels of analysis but collapse them into one level, and thus do not allow knowledge of the dialogue structure to be utilized in the analysis. In fact, if we reconsider the analysis made above of the utterance "When is the Windsor train?", we can identify a tension where the two levels interact. In particular, all the relationships (i.e., the arcs) in plans arise from a theory of problem solving, independent of linguistic actions. Thus we have arcs such as "effect of," "prerequisite," "part of," etc. However, there was one class of arcs indicated in the example as "knowledge necessary for" arcs. (In [Allen, 1979], these links were introduced by the knowledge inferences, *knowif*, *knowref*, etc.) These relate steps in a plan to prerequisite knowledge on the part of the actor, but were hard to motivate within the general problem solving theory. It is exactly at these links that the transition between communicative goals and task goals is made. In the new model the utterance "When does the Montreal train leave?" would be recognized at the communicative goal level as a *bid goal* to obtain information (about the departure time). This analysis allows the task level analysis to recognize the user's ultimate goal of boarding the train.

The overall architecture of the system is depicted in Figure 2. Included as well is the generative side of the system which is not currently being implemented. Using this figure, let us consider what the system behavior would be if the user had said only the opening utterance of Dialogue 1.

Task Reasoning: System Planning

Recognition of User's Task	→	Obstacle Detection	→	Generation of System Goals

^	|	|	^
observed discourse goals	expected discourse goals	desired discourse goals	pursued discourse goals
|	v	v	|

Communicative Goal Reasoning

Recognition of Communicative Act	Planning Communicative Acts

^	|	|	^
observed linguistic acts	expected linguistic acts	desired linguistic acts	pursued linguistic acts
|	v	v	|

Linguistic Reasoning

Parser	Linguistic Generator

^	v
Input	Output

Figure 2.

The utterance "Could you mount a magtape for me?" could be analyzed at the linguistic level as either a yes/no question or an indirect request. The indirect request interpretation arises because of the idiomatic nature of the utterance. Note that since the communicative goal reasoner is able to take the literal and infer the indirect act as well, the indirect request need not be recognized at the linguistic level. These observed linguistic acts are sent to the communicative goal level. Using this input, the communicative goal reconized is a *bid goal* to mount the magtape, which is sent to the task reasoner. The task reasoner analyzes the communicative goal and produces a plan for the task. In this simple example, it could simply introduce a top-level mutual goal of mounting the tape.

This goal can then be expanded by the task reasoner and the resultant plan inspected for obstacles. Assuming the user says nothing further, there is an obstacle in the task plan, for the system does not know which tape to mount. This generates a system goal to identify the tape parameter, which is sent to the communicative goal reasoner. A speech act (or acts) is planned that will lead to accomplishing the goal and which obeys the constraints on well-formed discourse. This would be sent to the linguistic level where a response would be generated, such as "which tape?"

The interactions are considerably simplified in the above example. In order to be able to recognize sentence fragments, and to recognize linguistic clues as to the discourse structure, the parser must send partial descriptions as the utterance is being analyzed. Example messages could be "a noun phrase referring to a tape was mentioned," or "the utterance was preceded with a 'but'" (indicating topic change). One design objective is to make it possible for the system to generate a reasonable response even if the parser fails to generate a complete analysis of the utterances. To allow such behavior we view each of the levels of analysis as running in parallel. In the implementation, each level is implemented by one or more processes and the levels interact using message passing (e.g., [Feldman, 1979]). Thus, although we have separated out various stages of analysis, the utterances are not processed by one stage at a time in sequence.

In the actual dialogue we saw the user identify the tape before the system had a chance (or possibly realized the need!) to generate a request to identify it. It is not plausible to allow the system to ignore such new information and generate the response anyway. On the other hand, some system responses, especially those that correct a bad assumption on the part of the user, should be generated anyway and the input effectively ignored. To make such a decision the system needs to know both the import of the user's new utterance and the goals underlying its response to the original utterance.

Our initial solution to this problem is to have the linguistic generation level check with the task level just before the response is actually generated to see if the goal that motivated the response is still valid. Thus the task level of the system is responsible for some coordination of behavior between the other levels.

Finally, each module is connected to a knowledge base of facts. We have developed a representation language which is a variant of FOPC that allows knowledge to be structured in a manner akin to semantic networks. Associated with the representation is a specialized limited inference mechanism that mimics the role of a network matcher and provides the system with general inference behavior such as the inheritance of properties and limited reasoning about coreference, time, and beliefs. This will be considered in detail in Section 5.

4. A Closer Look at the Interfaces

4.1 The Communicative Level/Task Level Interface

Given that the new system splits the analyses of intention into two levels, the question arises as to what are the high-level goals at each, and how do they relate to each other. The high-level goals at the task level are dependent on the domain, but correspond to the high-level goals in the earlier system. The high-level communicative goals were not present previously, and must satisfy two constraints. First, they must reflect the structure of English dialogue. Second, though, they must be useful as input to the task level reasoner. In other words, they must specify some operation (e.g., introduce goal, specify parameter) that indicates how the task level plan is to be manipulated.

Our initial set of high-level communicative goals is based on the work of Mann, Moore and Levin [1977]. In their model, conversations are analyzed in terms of the manipulation of goals in the task domain. Thus, typical communicative goals are reflected by the actions:

Bid-Goal--introduction of a task goal for adoption by the hearer;

Accept-Goal--acceptance by the hearer of a bid goal;

Parameter Specification--identification of a parameter in an already accepted task;

Termination--end of a discussion and pursuit of an already accepted goal.

These are suitable for our analysis, for each specifies some specific operation that the task level reasoner should perform. Of course, since the task level reasoner is a general plan recognizer as well, it may infer beyond the immediate effect of the specific communicative action inferred at any one stage. For example, if a goal is bid to mount a tape, the system might infer that the user has a higher-level goal of restoring a file, or possibly stacking up a file.

We have specified these communicative goals as actions in our plan model, outlining their prerequisites, effects, and methods for accomplishing them. These tie in with the speech act analysis in the original system easily. Thus, using the same plan recognition algorithm as before, we can recognize the communicative goals.

Not all of these communicative actions are possible at any given time. For instance, at the start of a dialogue, one may either bid a goal or get the other agent's attention (a summons). In order to capture this knowledge we have a context-free grammar which has these communicative acts as terminals, along the lines of Horrigan [1977]. The grammar indicates what acts are legal at any particular time for both participants. In order to produce such a grammar, we needed to extend the set of communicative acts to include acts such as summoning attention, acknowledgments, etc., which are included in [Horrigan, 1977]. This model is currently being implemented and tested on some sample dialogues, including Dialogue 1. We are currently considering incorporating a more general model of discourse that can handle a wider range of dialogues, including topic change, clarification dialogues, and repair.

4.2 The Communicative Goal/Linguistic Level Interface

One of the results of the previous system was that some utterances consisting of a single noun phrase could be understood appropriately. The context was sufficient to identify one plausible plan for the speaker. We hope to generalize this result to ungrammatical utterances. As the linguistic analysis progresses, it can notify the communicative goal level of the various noun phrases that appear as they are analyzed. This allows the other levels to start analyzing the speaker's intentions before the entire sentence is linguistically analyzed. Thus, sometimes an interpretation may be found even if the linguistic analysis eventually "fails" to find a complete sentence. (Failure is not quite the correct word here, since if the utterance is understood, whether it was "correct" or not becomes uninteresting.)

In addition, the rest of the system may be able to provide the linguistic level with strong enough expectations as to the content of the utterance that it is able to construct a plausible analysis of what was said.

We are currently investigating what other partial information could be useful for the rest of the system. One area that is obvious is the recognition of *clue* words to the discourse structure [Reichman, 1978]. For example, if the next user utterance begins with the word "but," this gives a clue as to what communicative goal the user is performing. In particular, the system should expect the user to modify the current topic in some way. Similarly, if an utterance contains the word "please," then the intent behind the utterance will involve a *request* at some level of analysis.

5. Issues in Knowledge Representation

One of the more important first tasks in designing the system was to specify a system-wide language in which facts could be expressed and transmitted in messages. One of the methodological goals in this development was not to introduce any constructs into this language until they were rigourously defined. We started with a standard version of the first order predicate calculus and have since introduced notational abbreviations and defined a wide range of predicates at two separate levels of analysis. The first level, corresponding to the epistemological level in [Brachman, 1979], consists of predicates that are used to define the structure of knowledge. The initial set of these has been determined by investigating what types of inferences we want to be able to do efficiently and automatically. Given these predicates and the set of desired inferences, we have defined a retrieval component acting on a knowledge base of facts. The current retriever implements such inferences as those that produce semantic network-like inheritance of properties. This work is considered in more detail in Section 5.2

The other level of analysis corresponds to the conceptual level of [Brachman, 1979]. At this level we have outlined basic theories of the structure of actions, events, plans, times, and beliefs. Using these theories, we then have specified hierarchies of actions and events, eventually arriving at predicates that are specific to the domain being modeled. Some of the theoretical underpinnings of this work are outlined in Section 5.2.

5.1 The Epistemological Primitives and the Retriever

Ever since Woods's [1975] "What's in a Link" paper, there has been a growing concern for formalization in the study of knowledge representation. Several arguments have been made that frame representation languages and semantic network languages are syntactic variants of the first-order predicate calculus (FOPC). The typical argument (e.g., [Hayes, 1979; Nilsson, 1980]) proceeds by showing how any given frame or network representation can be mapped to a *logically isomorphic* (i.e., logically equivalent when the mapping between the two notations is accounted for) FOPC representation. We emphasize the term "logically isomorphic" because these arguments have primarily dealt with the content (semantics) of the representations rather than their forms (syntax). Though these arguments are valid and scientifically important, there is another side to the story.

For the past two years we have been studying the formalization of knowledge retrievers as well as the representation languages that they operate on. This study has led to the conclusion that the form of a representation is crucial to the design of a retriever. We are designing a representation language in the notation of FOPC whose form facilitates the design of a semantic-network-like retriever.

Elsewhere [Frisch and Allen, 1982], we have demonstrated the utility of viewing a knowledge retriever as a specialized inference engine (theorem prover). A specialized inference engine is tailored to treat certain predicate, function, and constant symbols differently than others. This is done by building into the inference engine certain true sentences involving these symbols and the control needed to handle with these sentences. The inference engine must also be able to recognize when it is able to use its specialized machinery. That is, its specialized knowledge must be coupled to the *form* of the situations that it can deal with.

For illustration, consider an instance of the ubiquitous type hierarchies of semantic networks:

$$\begin{tikzcd} \text{FORDS} \\ \text{MUSTANGS} \arrow[u, "\text{subtype}"'] \\ \text{OLD-BLACK} \arrow[u, "\text{type}"'] \end{tikzcd}$$

By considering the types *FORDS* and *MUSTANGS* to be predicates, the following two FOPC sentences are logically isomorphic to the network:

(1.1) $\forall$ x MUSTANGS(x) $\rightarrow$ FORDS(x)

(1.2) MUSTANGS(OLD-BLACK)

However, these two sentences have not captured the form of the network, and furthermore, not doing so is problematic to the design of a retriever. The subtype and type links have been built into the network language because the network retriever has been built to handle them specially. That is, the retriever does not view a subtype link as an arbitrary implication such as (1.1) and it does not view a type link as an arbitrary atomic sentence such as (1.2).

In our representation language we capture the form as well as the content of the network. By introducing two predicates, *TYPE* and *SUBTYPE*, we capture the meaning of the type and subtype links. $TYPE(i,t)$ is true iff the individual i is a member of the type (set of objects) t, and $SUBTYPE(t_1,t_2)$ is true iff the type t_1 is a subtype (subset) of the type t_2. Thus, in our language, the following two sentences would be used to represent what was intended by the network:

(2.1) SUBTYPE(FORDS,MUSTANGS)

(2.2) TYPE(OLD-BLACK,FORDS)

It is now easy to build a retriever that recognizes subtype and type assertions by matching predicate names. Contrast this to the case where the representation language used (1.1) and (1.2) and the retriever would have to recognize these as sentences to be handled in a special manner.

But what must the retriever know about the SUBTYPE and TYPE predicates in order that it can reason (make inferences) with them? There are two assertions, (A.1) and (A.2), such that {(1.1),(1.2)} is logically isomorphic to {(2.1),(2.2),(A.1),(A.2)}. (Note: throughout this paper, axioms that define the retriever's capabilities will be specially labeled A.1, A.2, etc.)

(A.1) $\forall\ t_1,t_2,t_3\ \mathrm{SUBTYPE}(t_1,t_2) \wedge \mathrm{SUBTYPE}(t_2,t_3) \rightarrow \mathrm{SUBTYPE}(t_1,t_3)$

(*SUBTYPE* is transitive.)

(A.2) $\forall\ o,t_1,t_2\ \mathrm{TYPE}(o,t_1) \wedge \mathrm{SUBTYPE}(t_1,t_2) \rightarrow \mathrm{TYPE}(o,t_2)$

(Every member of a given type is a member of its subertypes.)

The retriever will also need to know how to control inferences with these axioms, but this issue is not taken up in this paper.

The design of a semantic-network language often continues by introducing new kinds of nodes and links into the language. This process may terminate with a fixed set of node and link types that are the knowledge-structuring primitives out of which all representations are built. Others have referred to these knowledge-structuring primitives as epistemological primitives [Brachman, 1979], structural relations [Shapiro, 1979], and system relations [Shapiro, 1971]. If a fixed set of knowledge-structuring primitives is used in the language, then a retriever can be built that knows how to deal with all of them.

The design of our representation language very much mimics this approach. Our knowledge-structuring primitives include a fixed set of predicate names and terms denoting three kinds of elements in the domain. We give meaning to these primitives by writing domain-independent axioms involving them. A retriever has been built that reasons with these axioms and thus knows how to deal with all the primitives of our language. Thus far in this paper we have introduced two predicates (*TYPE* and *SUBTYPE*), two kinds of elements (individuals and types), and two axioms ((A.1) and (A.2)).

This type of analysis can be continued to introduce roles, distinguished types, and limited forms of equality [see Allen and Frisch, 1982].

The important point to notice here is that once we have selected our predicates and given the axioms defining them, we have a precise characterization of what inferences we would like the retrieval component to perform. We have used this approach to define a prototype knowledge base retrieval mechanism that is currently being used in the system. It is implemented in a Horn clause theorem prover and provides one with approximately the same capabilities as the partitioned networks of Hendrix [1979], and makes retrievals reasonably efficiently.

5.2 Formal Aspects of the Conceptual Level of Representation

An important part of this research over the last two years has been the investigation of some basic issues in representation. In particular, the existing models of action were inadequate to represent many of the concepts talked about in even simple dialogues, as well as being inadequate for a more general plan reasoning. This problem was mainly caused by an inadequate treatment of time in existing knowledge representations. The other major problem was the precise specification of a representation of belief that did not lead to theoretical difficulties. Progress has been made on all of these issues.

An interval-based temporal logic has been defined [Allen, 1981a] and is currently being incorporated into our knowledge representation. Relationships between intervals are maintained in a hierarchical manner and an inference process based on constraint propagation has been developed and implemented. This representation is notable in a few areas:

-- It allows one to efficiently represent the present moment (i.e., "now") so that it can be continually updated without making major changes to the knowledge base.

-- It is designed using relative information about how intervals are related. Thus it doesn't depend on a date line which is often found in temporal representations. This is particularly important in a dialogue system for most temporal information does not have a precise time.

-- It allows time intervals to extend indefinitely into the past or future, and supports a limited type of default reasoning.

This representation of time has been used to produce a general model of events and actions [Allen, 1981b]. Rather than concentrating on how actions are performed, as is done in the problem-solving literature, this work examines the set of conditions under which an action or event can be said to have occurred. In other words, if one is told that action A occurred, what can be inferred about the state of the world?

Consider an example investigated in detail in [Allen, 1981b]. What are the conditions under which one might say that an actor hid a book from another actor? Certainly, this can't be answered in terms of the physical actions the actor did, for the actor might have hid the book by

-- putting it behind a desk;

-- standing between it and the other agent while they are in the same room; or

-- calling a friend and getting him to do one of the above.

Furthermore, the actor might hide the object by simply not doing something s/he intended to do. For example, assume Sam is planning to go to lunch with Carole after picking Carole up at her office. If, on the way out of his office, Sam decides not to take his coat because he doesn't want Carole to see it, then Sam has hidden the coat from Carole. Of course, it is crucial here that Sam believed that he normally would have taken the coat. Sam couldn't have hidden his coat by forgetting to bring it.

This example brings up a few key points that may not be noticed from the first three examples. First, Sam must have intended that Carole not see the coat. Without this intention (i.e., in the forgetting case), no such action occurs. Second, Sam must have believed that it was likely that Carole would see the coat in the future course of events. Finally, Sam must have acted in such a way that he then believed that Carole would not see the coat in the future course of events. Of course, in this case, the action Sam performed was "not bringing the coat," which would normally not be considered an action unless it was intentionally not done.

I claim that these three conditions provide a reasonably accurate definition of what it means to hide something. They certainly cover the four examples presented above. It is also important to note that one does not have to be successful in order to have been hiding something. The definition depends on what the hider believes and intends at the time, not what actually occurs. However, the present definition is rather unsatisfactory, as many extremely difficult concepts, such as belief and intention, were thrown about casually.

In the last two years, we have developed a model of belief by viewing BELIEVE as a predicte between an agent and a description of a sentence. To do this, we must introduce quotation into the logic. Thus the assertion "John believes Sam lives on 4th Street" would be expressed as

BELIEVE(JOHN,"LIVES(SAM,4thSTREET)").

Introducing quotation into a logic does not cause any difficulties until one tries to relate the quoted formula to the formula it names. To do this, we need a truth predicate, and an axiom such as: for any sentence α

(*) TR("α") <=> α.

Thus,

TR("LIVES(SAM,4thSTREET)") <=> LIVES(SAM,4thSTREET).

Unfortunately, such an axiom leads to paradoxes. Perlis [1981], however, showed that one can define a truth scheme that intuitively gives us the behavior above but which is provably consistent. There is not the space to examine this here, but suffice to say that (*) does not get us into trouble unless α contains a negation outside a "Tr" predicate.

Using this formalism, we can safely introduce the BELIEVE predicate and examine its behavior. One of the initial difficulties concerns representing the fact that someone knows something that the believer does not know. For instance, if it is not known where Sam lives, we would still like to be able to represent the fact that John knows where Sam lives. This is typically handled by quantifying in. Thus we get a formula such as

(**) $\exists$ x BELIEVE(JOHN,"LIVES(SAM,x)").

I have been deliberately loose here about quotation. Actually the variable x ranges over quoted expressions and must not be quoted. So we need a more elaborate quotation scheme that gives us the abilities of Quine's corner quotes. Leaving these details aside, however, the above formula does not capture the required knowledge. Presumably, everyone believes that Sam lives where Sam lives, so the description "the place where Sam lives" satisfies (**) but does not capture that John knows where Sam lives.

One way out of this problem is to assume there is a *standard name* for every object (e.g., Moore [1975]). This is inadequate, however, for the name that will satisfy the above knowledge changes as the context changes. For example, if John were a customs officer at the border, the description "Rochester" would be enough to claim that John knows where Sam lives. If John were a friend going to Sam's house, however, directions to the house (e.g., an address) would be required. Thus to solve this problem we need to be able to assert what descriptions are useful for what task, and then knowing what something is depends on what task is being considered.

Within a logic with quotation, however, predicates that operate on the syntactic form of formulas are perfectly acceptable, and one can specify exactly what form of description is necessary for any task. Thus for JOHN the customs officer at the border, he knows where Sam lives if

$\exists$ x BELIEVE(JOHN,"LIVES(SAM,x)") & CITY-NAME(x)

where CITY-NAME is a predicate on expressions and is true if x is the proper name of a city. The interested reader should see [Haas, 1982] for further details.

One problem with quotation schemes that is also solved by Haas is that if one simulates another's reasoning by simulating inference rules on syntactic formulas, the length of the simulation with respect to the simulated reasoning grows exponentially with the depth of nesting of beliefs. An approach that avoids this involves collecting all the beliefs of the agent in question into a separate "data base" and then running the inference rules on only those facts. This technique, however, appears not to be able to handle beliefs that involve quantifying in or to use knowledge involving disjunctions of beliefs. Techniques have been devised to remedy these problems. By introducing the concept of *dummy constants* along the lines of [Cohen, 1978], we can handle the quantifying in case. Haas [1982] presents a rigorous tratment of these issues. Since the simulation technique is just another proof rule in a general inference system, disjunctions can be handled using the standard techniques.

References

Allen, J.F., "A plan-based approach to speech act recognition," Ph.D. thesis, Computer Science Dept., U. Toronto, 1979.

Allen, J.F., "An interval-based representation of temporal knowledge," *Proc.*, 7th Int'l. Joint Conf. on Artificial Intelligence, Vancouver, B.C., August 1981a.

Allen, J.F., "What's necessary to hide?: Reasoning about action verbs," *Proc.*, 19th Annual Meeting, Assoc. for Computational Linguistics, 77-81, Stanford U., 1981b.

Allen, J.F. and C.R. Perrault, "Analyzing intention in utterances," *Artificial Intelligence 15*, 3, 1980.

Brachman, R.J., "Taxonomy, descriptions, and individuals in natural language understanding," *Proc.*, 17th Annual Meeting, Assoc. for Computational Linguistics, 33-37, UCSD, La Jolla, CA, August 1979.

Bruce, B. and D. Newman, "Interacting plans," *Cognitive Science 2*, 3, 195 233, July September 1978.

Carbonell, J.G., "POLITICS: Automated ideological reasoning," *Cognitive Science 2*, 1, 27-51, 1978.

Cohen, P.R., "Planning speech acts," Ph.D. thesis and TR 118, Dept. of Computer Science, U. Toronto, 1978.

Feldman, J.A., "High-level programming for distributed computing," *CACM 22*, 6, 363-368, June 1979.

Grosz, B.J., "Discourse Knowledge," Section 4 in D.E. Walker. *Understanding Spoken Language.* New York: North-Holland, 1978.

Haas, A., "Mental states and mental actions in planning," Ph.D. thesis, Computer Science Dept., U. Rochester, 1982.

Hendrix, G.G., "Encoding knowledge in partitioned networks," in N.V. Findler (Ed.). *Associative Networks.* New York: Academic Press, 1979.

Horrigan, M.K., "Modelling simple dialogues," *Proc.*, 5th Int'l. Joint Conf. on Artificial Intelligence, MIT, 1977.

Johnson, P.N. and S.P. Robertson, "MAGPIE: A goal-based model of conversation," Research Report #206, Computer Science Dept., Yale U., May 1981.

Mann, W.C., J.A. Moore, and J.A. Levin, "A comprehension model for human dialogue," *Proc.*, 5th Int'l. Joint Conf. on Artificial Intelligence, MIT, 1977.

Moore, R.C., "Reasoning about knowledge and action," Ph.D. thesis, MIT, February 1979.

Perlis D., "Language, computation, and reality," Ph.D. thesis, Computer Science Dept., U. Rochester, 1981.

Perrault, C.R. and J.F. Allen, "A plan-based analysis of indirect speech acts," *J. Assoc. Comp'l. Linguistics 6*, 3, 1980.

Reichman, R., "Conversational coherency," *Cognitive Science 2*, 1978.

Walker, D.E. *Understanding Spoken Language*. New York: North-Holland, 1978.

Wilensky, R., "Understanding goal-based stories," Ph.D. thesis, Yale University, 1978.

Woods, W.A., "What's in a link: Foundations for semantic networks," in D.G. Bobrow and A. Collins (Eds). *Representation and Understanding: Studies in Cognitive Science*. New York: Academic Press, Inc., 1975.

END

FILMED

9-83

DTIC